

COMPARATIVE ANALYSIS ON HOMOMORPHIC IMAGE ENCRYPTION ALGORITHMS

PROJECT REPORT

Submitted by

NAME	REG NO	MOBILE	EMAIL
KUSHAGRA SINGH	18BCE0017	9810952845	kushagra.singh2018@vitstudent.ac.in
PAVAN SIDDHARTH E	18BCE0044	6379063582	epavan.siddharth2018@vitstudent.ac.in
ADHIL MOHAMMED	18BCE0056	9840761863	adhil.mohammed2018@vitstudent.ac.in
HARISH BHARADWAJ	18BCE0078	9003496945	harishbharadwaj.s2018@vitstudent.ac.in
SAMYAK JAIN	18BCE0083	9969187312	samyak.jain2018@vitstudent.ac.in

B.Tech.

in

Computer Science and Engineering

School of Computer Science & Engineering

®



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

INDEX

ABSTRACT

1. Introduction

1.1. Theoretical Background

1.2. Motivation

1.3. Aim of the proposed Work

1.4. Objective(s) of the proposed work

2. Literature Survey

2.1. Survey of the Existing Models/Work

2.2. Summary/Gaps identified in the Survey

3. Overview of the Proposed System

3.1. Introduction and Related Concepts

3.2. Framework, Architecture or Module for the Proposed System

3.3. Proposed System Model (Mathematical Modeling)

4. Proposed System Analysis and Design

5. Results and Discussion

6. Conclusion

7. References

ABSTRACT

Privacy of users is very important especially with the rise in popularity and usage of cloud computing. One important format of data that is being used a lot nowadays is digital image and there are so many services provided in cloud computing that perform lots of transformations on those images. To protect their data, there is an encryption scheme - homomorphic encryption. Consider a user has to make use of an image processing service provided over the cloud. The user just uploads the image without encrypting it and now the service provider can see and has access to the image, which means it is a breach of the user's privacy. This is where homomorphic encryption comes in. The user's image is first encrypted with a secret key and after encryption, whatever transformation has to be performed on the image, can be performed on the encrypted image itself without needing to reveal what the actual image is. After this step, when decryption is performed the result is an image with the transformation applied on the original, unencrypted image. We are going to perform some image processing services using the homomorphic encryption scheme.

Key Words: Homomorphic Encryption, Image Processing

1. Introduction

1.1. Theoretical Background

As of today, almost all homomorphic encryption techniques work only for text and not images. ElGamal and the unpadded RSA schemes work only when the transformation function consists only of multiplication operations. Micali and Goldwasser systems work only when the transformation function consists only of addition operation. So all the above mentioned systems are partial solutions for homomorphic encryption. A full homomorphic system was proposed by Gentry but the encryption took a lot of time even for a small image and hence the algorithm was not feasible. The earlier techniques were developed only to operate on integer data but now we need floating point operations for image processing which means changes have to be made to existing algorithms before they can be applied on floating point numbers.

1.2. Motivation

Many images are in circulation nowadays and some of them on the internet are private and confidential. So, we need to make sure that the privacy of the user's data is not compromised. A homomorphic encryption scheme can be very helpful in this scenario because then the users only need to upload their encrypted images and when the service provider has performed the specific transformation operation on that encrypted image without looking at the contents of the image, the user can decrypt it with his key and then the transformation would have been applied on his image. The privacy of the user's data is safeguarded in this process.

1.3. Aim of the proposed Work

The aim of our project is to demonstrate the image processing techniques on the encrypted images using 2 different homomorphic encryption algorithms and to compare the performance and security of the homomorphic encryption algorithms.

1.4. Objective(s) of the proposed work

We perform image processing operations like finding the complement of an image, increasing the brightness of an image using two homomorphic encryption algorithms. We perform histogram analysis for the original and encrypted images and correlation analysis (horizontal and vertical correlation) for the original and encrypted images. We also measure the performance of the algorithm while performing each transformation by measuring the time taken to perform encryption, to apply the transformation and to decrypt the image back.

2. Literature Survey

2.1. Survey of the Existing Models/Work

S.No	PAPER NAME	AUTHOR	METHODOLOGY	CONCLUSION
1	Homomorphic encryption as a service for outsourced images in mobile cloud computing environment	Mouhib Ibtihal; El Ouadghiri Driss; Naanani Hassan	This paper mainly focuses on securing the outsourcing of the images. To do this they proposed an architecture that is securely composed of 2 clouds, one of which is a private cloud for encryption/decryption and the other is for storage. They used OpenStack for the first cloud using Paillier's homomorphic cryptosystem, the test for which was done by applying DWT	Paper proposed an architecture for secured outsourced images in mobile cloud computing. It is based on encryption used as a service which has a private cloud for encryption and decryption. Additive homomorphic encryption system based on paillier's cryptosystem was implemented and its functionality was tested.
2	Cryptolmg: Privacy Preserving Processing Over Encrypted Images	M. Tarek Ibn Ziad; Amr Alanwar; Moustafa Alzantot; Mani Srivastava	This paper describes the design and implementation of Cryptolmg which is a library of image processing operations for preservation of encrypted images. It uses homomorphic encryption to allow users to use image processing operations to servers remotely without privacy concerns. This paper implements its library as an extension to the computer vision library OpenCV.	Secured operations like spatial filtering, image adjustment, histogram equalization, edge detection, edge sharpening and morphological operations were safely outsourced to 3 rd party servers and no privacy issues were recorded. Paper presented how these operations were implemented with less time overhead and with a single communication round.

3	Homomorphic Image Encryption	Sachin Rana, Om Jadhav , Shivam Rajput, Pranjal Bhansali, Varshapriya Jyotinagar	Extended Paillier homomorphic encryption algorithm which was usually text-based to images. They performed the encryption with RGB images	Only image addition was feasible to use in applications. Image transformation was not feasible and took lot of time
4	An Efficient Secret Key Homomorphic Encryption Used in Image Processing Service	Pan Yang, Xiaolin Gui, Jian An, and Feng Tian	This paper focuses on improving Gentry homomorphic asymmetric encryption algorithm to symmetric encryption algorithm. The author has also performed cryptanalysis on the algorithm.	The security level was the same as the normal Gentry algorithm. The time to perform the encryption is very high hence not suitable to use in applications.
5	Paillier Cryptosystem based Mean Value Computation for Encrypted Domain Image Processing Operations	Mohsin Shah, Weiming Zhang, Honggang Hu, and Nenghai Yu	This paper shows how image processing operations like mean value computation and division operation can be performed in the homomorphic encrypted domain with Paillier cryptosystem.	Image processing operations like filtering and masking are performed in the encrypted domain without any pre processing . Also the privacy of the images are preserved while performing these operations.
6	Secure Image Watermarking in a Compressed SPIHT Domain Using Paillier Cryptosystem	Ritu Gupta, Anurag Mishra, Sarika Jain	The paper performs homomorphic Paillier encryption on 12 different RGB images as well as a binary watermark and real time embedding of the watermark onto the images. Five different image processing attacks are carried out on the encrypted and watermarked images to	After decryption, when the extracted watermark is compared with the original one after performing all the image processing attacks, it is concluded that best results are achieved when the key size is taken as 2048.

			calculate the robustness of the watermarking procedure.	
7	Distributed Image Encryption Based On a Homomorphic Cryptographic Approach	Wade, M. I., Chouikha, M., Gill, T., Patterson, W., Washington, T. M., and Zeng, J.	The authors have proposed a methodology to use a homomorphic function in such a way to encrypt an image to produce more than one cipher image. The R, G and B components are extracted by decomposing each channel image into a sum of several pixel intensity sub-values which then produces images with multiple component channels. These are separately encrypted with the help of the same encryption key.	The algorithm is able to produce highly secure images, usually applied on those that contain confidential data. The main drawback that arises from this algorithm is that real-time applications may be possible only if faster micro-processors are used which is a major drawback as the time to encrypt or decrypt on an average computer will be too high.
8	Reversible data hiding in encrypted images with somewhat homomorphic encryption based on sorting block-level prediction-error expansion	Xiong, L., and Dong, D.	The authors of the research paper have proposed a methodology of reversible data hiding scheme for encrypted images with SHE (Somewhat Homomorphic Encryption) based on sorting block-level error expansion (SBPEE). The image is divided into 2x2 blocks where SHE is used to protect the content of the image. During encryption, the data hider embeds the secret data using SBPEE, which is based on pixel redundancy within each block. If the receiver only has the data hiding key, the additional data can be extracted but if they also	The performance of this method is better than existing methods when a low embedding rate is used. The limitation of this method is that when using a high embedding rate, the image quality gets low along with a reduction in the security of the scheme.

			have the encryption key, the original image can also be recovered without loss.	
9	Red Green Blue Image Encryption Based on Paillier Cryptographic System	Mamadoul Wade Henry C.Ogworonjo MadihaGul Mandoye Ndoeye Mohamed Chouikha Wayne Patterson	This paper presents a new application of the Paillier cryptographic system. It provides encryption of red green blue (RGB) images. The image is first split into RGB color channels and the Paillier encryption function is applied to each of the channel's pixel intensity values. The next step is to combine the encrypted image and compress if necessary before transmitting the information through an insecure network.	Results from these tests show that the proposed image encryption scheme produce cipher images that can resist these security attacks, as well as provide high quality recovered images
10	Elliptic curve ElGamal based homomorphic image encryption scheme for sharing secret images	Li Li Ahmed A Abd El-Latif Xiamu Niu	This paper proposes an encryption scheme with an enhanced Elliptic Curve ElGamal (EC-ElGamal) based on additive homomorphism. Its application mainly include sharing of images securely over insecure network. The proposed algorithm provides shorter key which enables better performance than schemes based on RSA or ElGamal. It has a lower computation overhead in image decryption comparing with the method that uses other additively homomorphic property in EC-ElGamal. Elliptic curve parameters are selected to resist the Pohlig–Hellman, Pollard's-rho, and Isomorphism attacks.	It achieves the same security level with smaller key size in contrast with RSA and ElGamal. The efficiency and security are exploited by the experimental results and analysis which concluded better performance than those methods based on RSA, ElGamal, and current work based on EC-ElGamal.

2.2. Summary/Gaps identified in the Survey

One of the major limitations of why it is still not applicable to use in any application is the time taken to execute. Till now all homomorphic image encryption algorithms that have been proposed are text-based that have been extended to images. This means that the operations are done at pixel-by-pixel level. While for small images this does not matter but for large images these operations take a lot of time since there are a lot of pixels. This is the reason it is not used in applications a lot. So, either we have to compromise on security by reducing the key size i.e., the encryption values are small so operations could be done faster thus reducing the execution time or we have to compromise on time by providing security. For applications these two factors i.e., time and security must go hand in hand. If one of them fails the users won't be satisfied. The other reason why homomorphic encryption is not used is because the homomorphic property for fast algorithms is only supported for addition and multiplication operations. This means that for complex operations like log or sine or cosine we have to reduce them in linear form before applying any operation. Although it is possible for some of the functions to be reducible to linear form, it is not possible for all functions. Recently more and more researchers are trying to come up with ways to make different functions into linear form.

One of the solutions will be to upgrade the hardware of the cloud service provider. This solution is not a very good one since it incurs a lot of cost. And even though if we upgrade it, it is still not feasible to complete a ton of user requests that will be coming, in a short period of time. Other solution that could solve the problem is to use an algorithm that satisfies homomorphic properties on matrices rather than a single number. This will reduce execution time drastically. This is well suited for encrypting images. Since operations can be done in a single go and since security is also not compromised this will be well suited for applications.

3. Overview of the Proposed System

3.1. Introduction and Related Concepts

Homomorphic encryption is a type of encryption with one additional feature that evaluates capability for computing over encrypted information without having access to the secret key. This results in the computation being encrypted. Homomorphic encryption is an extension to either public-key or symmetric-key cryptography. can be viewed as an extension of either symmetric-key or public-key cryptography. Homomorphic refers to the encryption and decryption functions that can be thought of as homomorphisms between plaintext and ciphertext spaces.

This includes more than one type of encryption scheme that performs multiple classes of computations of encrypted data. Fully homomorphic encryption, partially homomorphic, somewhat homomorphic and leveled fully homomorphic are some pf the common types of homomorphic encryptions:

- Fully homomorphic encryption enables the computation of arbitrary circuits built of multiple types of gates of unlimited depth, and is the most perfect notion of homomorphic encryption.
- Partially homomorphic encryption includes schemes that support the computation of circuits having only one kind of gate, for example, addition or multiplication.
- Somewhat homomorphic encryption schemes compute two kinds of gates, but only for a small part of circuits.
- Leveled fully homomorphic encryption covers the computation of arbitrary circuits built of multiple kinds of gates of limited or pre-determined depth.

3.2. Module for the Proposed System

Table 1. The Paillier cryptosystem

Step	Description
Key Generation	Let $L(x)$ be the largest integer v greater than zero such that $x \leq 1 \leq vn$. Choose two large primes p and q , then set $n = pq$, and $\lambda = \text{lcm}(p-1, q-1)$. Select an integer g , where $0 \leq g \leq n^2$ such that $\gcd(L(g^\lambda \bmod n^2), n) = 1$. The public key is (g, n) and the private key is (p, q) .
Encryption	The encryption of a plaintext $m \in \mathbb{Z}_n$ given the public key is $E(m) = g^m r^n \bmod n^2$, where r is a random integer in \mathbb{Z}_{n^2} .
Decryption	The decryption of a ciphertext $c \in \mathbb{Z}_{n^2}$ given the private key is $D(c) = L(c^\lambda \bmod n^2) \times (L(g^\lambda \bmod n^2))^{-1} \bmod n$.

Table 2. Efficient Homomorphic Medical Image Encryption Algorithm

Step	Description
Key Generation	KeGn(x) \rightarrow (key, p) Let p be a big prime number(public). Let q be a big prime number such that $p \gg q$ Let a be a random number (set of integers under p)
Encryption	E (key, w, g) = ag(wq+i) mod p Where: i is the input matrix. g is a small positive integer (cipher text degree) w is a big random number such that $ w + q < p $
Decryption	D (key, c, g) = (c\timesa-g mod p) mod q Where: c is the cipher (encrypted matrix) a ^{-g} is small integer (multiplicative inverse of a ^g)

3.3. Proposed System Model (Mathematical Modeling)

Paillier's cryptosystem:

Paillier's cryptosystem has the property of supporting homomorphic properties and follows non-deterministic encryption. One of the main characteristics of the Paillier's cryptosystem is its ability to support homomorphic properties. It also undergoes nondeterministic encryption

- Homomorphic additive property

If product of two ciphertexts is taken, sum of their original numbers will be decrypted,

$$D(E(P, r_1) \cdot E(Q, r_2) \bmod n^2) = (P+Q) \bmod n, \text{ where } P \text{ and } Q \text{ are real numbers.}$$

The product of a number with power g and the cipher text on decryption results in the sum of the given numbers,

$$D(E(P, r_1) \cdot g^Q \bmod n^2) = (P+Q) \bmod n.$$

- Homomorphic multiplicative property

If an encrypted number is raised to the power of some number, product of the two numbers is decrypted,

$$D(E(P, r_1)^Q \bmod n^2) = P \cdot Q \bmod n.$$

$$D(E(Q, r_2)^P \bmod n^2) = P \cdot Q \bmod n.$$

Efficient Homomorphic Medical Image Encryption Algorithm

Let c_1 and c_2 are the two cipher texts for the messages i_1 and i_2 . As mentioned above the parameters g_1 and g_2 are the small positive integers & w_1 and w_2 are the big random positive integers. p and q are the two big primes in which $p \gg q$, hence,

$$c_1 = a^{g_1(w_1q+i_1)} \bmod p$$

$$\text{And, } c_2 = a^{g_2(w_2q+i_2)} \bmod p$$

- Homomorphic Multiplicative property

$$\begin{aligned}
 & (c1 \times c2) \bmod p \\
 &= a^{g1(w1q+i1)} \bmod p \times a^{g2(w2q+i2)} \bmod p \\
 &= a^{g1+g2(w1w2q2+w1qi2+i1w2q+i1 \times i2)} \bmod p \\
 &= a^{g1+g2((w1w2q+w1i2+i1w2)q+i1 \times i2)} \bmod p
 \end{aligned}$$

- Homomorphic Additive property

$$(c1+c2) \bmod p = a^{g1(w1q+i1)} \bmod p + a^{g2(w2q+i2)} \bmod p$$

4. Proposed System Analysis and Design

The two main operations that we are performing on images to test the security of the images are:

1. Negation of Image

Negation of an image is the process of changing each pixel by a value that is obtained from subtracting the current pixel from the maximum intensity. This process is applied to the entire image to get the negation of the original image.

Formula: $\text{neg_pixel} = 255 - \text{curr_pixel}$

2. Change Brightness of Image

The brightness of an image can be changed by adding or subtracting a constant value from each pixel of the image. Adding a constant increases the brightness while subtracting leads to decreasing the brightness.

Formula: $\text{new_pixel} = \text{curr_pixel} + k$, where k is a constant.

Key Space Analysis

The key space of an encryption technique is the set of possible keys which can be utilized to encrypt data using the encryption technique. For a strong encryption algorithm, a large number of keys need to be tested in a brute force attack on that technique. The larger the size of a key, the harder it is to perform brute-force attacks.

Histogram Analysis

Histogram analysis is a technique where a histogram is plotted between the different intensity values found in the image to the number of pixels having each intensity value. It is a method to find the distribution of the pixels in an image. This analysis helps in giving the security level of an encrypted image. By performing histogram analysis of an encrypted image, it is observed that the streaks are distributed randomly across the histogram. It also tells if there is some part of the image that has not been encrypted properly. Hence, histogram analysis proves that a good cipher image generates a uniform histogram.

Correlation Analysis

Correlation Analysis is a method which is used to discover the relationship between two variables/pixels. A positive correlation determines how two variables increase or decrease together in parallel. Negative correlation indicates how one variable increases as another decreases. We perform the correlation analysis between the pixels of the original image and the encrypted image. This process is performed twice, once for vertically adjacent pixels, and then for horizontally adjacent pixels.

MSE and PSNR Analysis

Techniques like Mean-Square Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) are used to compare the compression quality of images. The MSE represents the cumulative squared error between the original image and the encrypted image. Similarly, the PSNR represents the measure of the peak error between the original and encrypted image. Higher values of PSNR indicate that the quality of the image is greater. These measures are used to indicate how much of image quality has been retained after performing encryption.

SSIM Analysis

The Structural Similarity Index Measure (SSIM) is a method that is used to predict the structural similarity between two images. This value ranges from 0 to 1. In our project, we apply the SSIM to the luminance value of each pixel in the original and encrypted image to measure the structural

similarity between neighboring pixels. High values of SSIM indicate that the images are very similar, while a value of 1 means that the images are identical.

5. Results and Discussion

5.1 Paillier's Homomorphic Algorithm

5.1.1 Histogram Analysis

By looking at this graph we could tell if the algorithm encrypts the image completely or is there any part of the image which is not properly encrypted. Below figure shows the histogram of the original image. We can say that the pixels are not distributed uniformly. Histogram of the encrypted image is also shown where it is uniformly distributed. It is fairly uniform which makes it difficult to extract the pixel's statistical nature of the plain image. By performing this histogram analysis it has been proven that a good cipher image generates a uniform histogram.

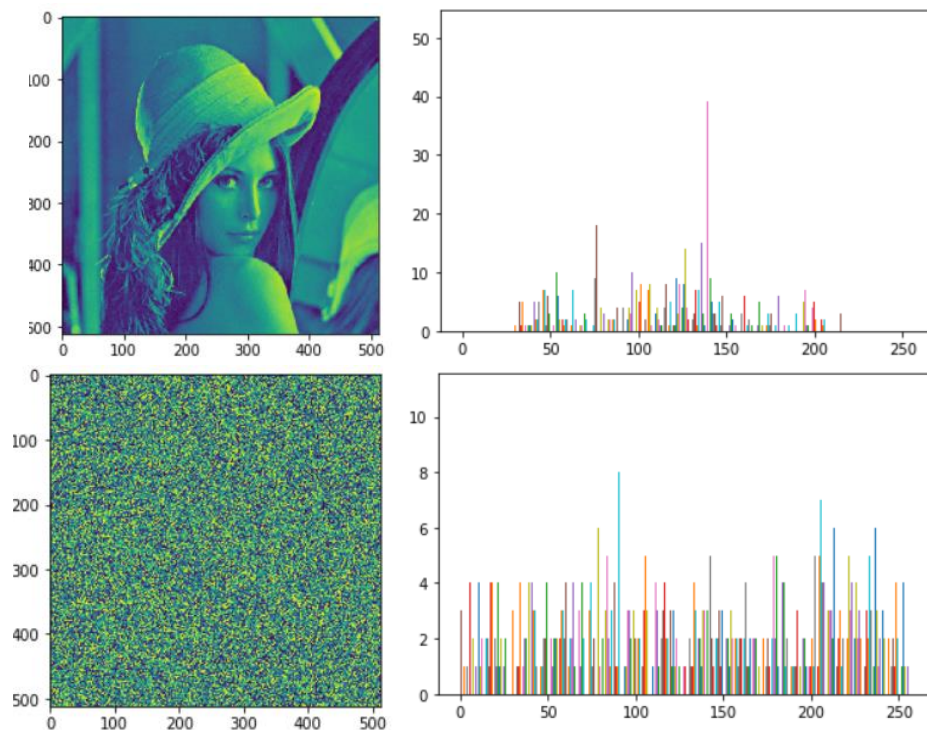


Fig: Histogram of Lena Image and Encrypted Lena Image for Paillier

5.1.2 Correlation Analysis

Below image indicates the high correlation existed between the adjacent pixels in the original image is now distributed in the resulting encrypted images.

5.1.2.1. Horizontal Correlation

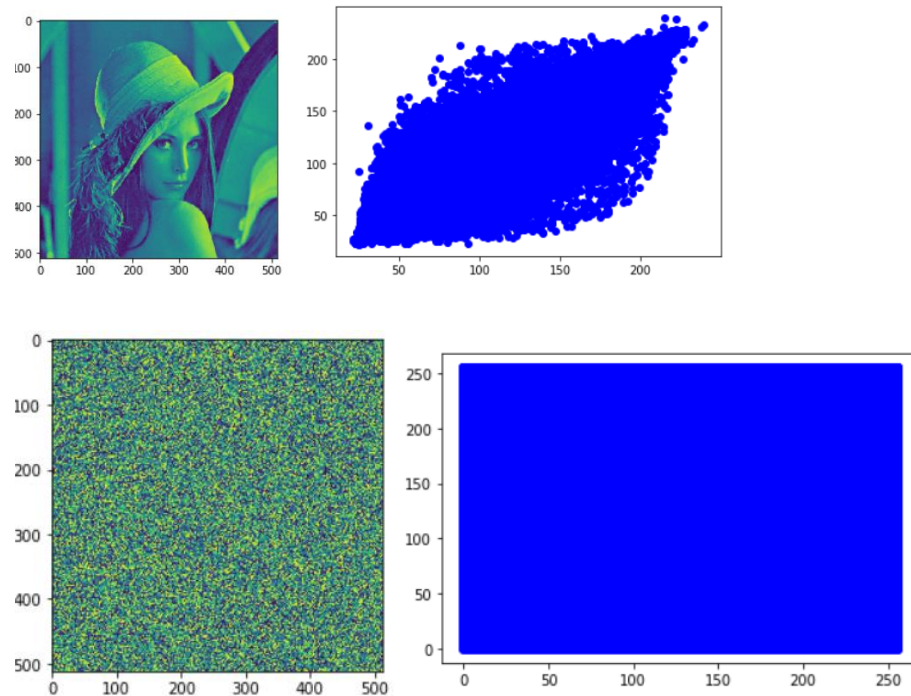
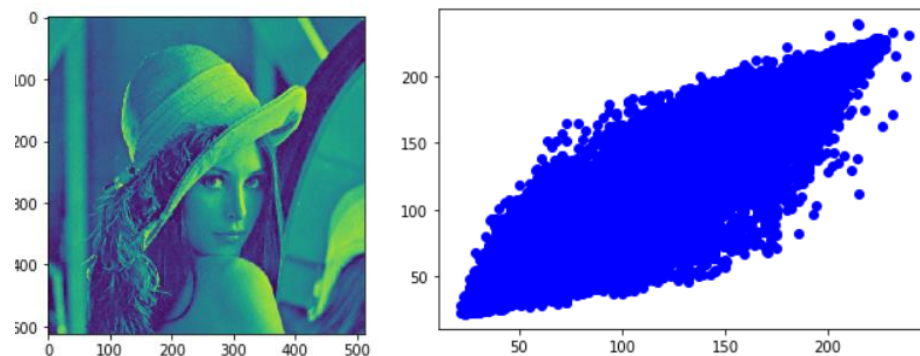


Fig: Horizontal correlation of Lena Image and Encrypted Lena image for Paillier

5.1.2.2 Vertical Correlation



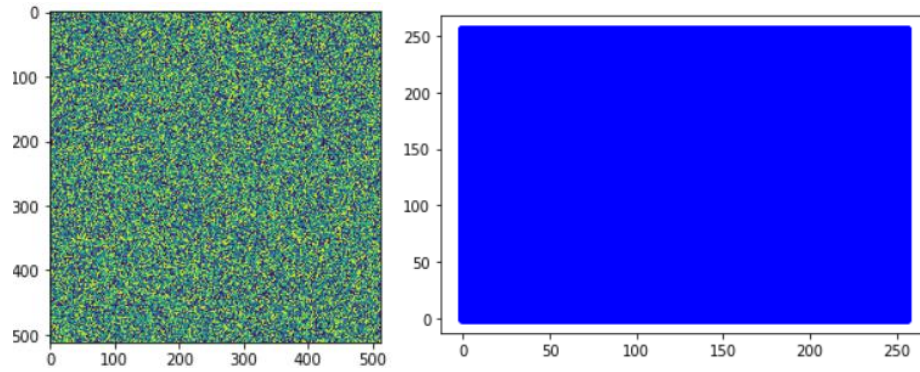


Fig: Vertical Correlation of Lena Image and Encrypted Lena image for Paillier

5.1.3. Intensity Transformation

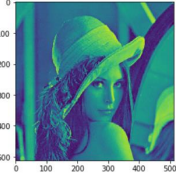
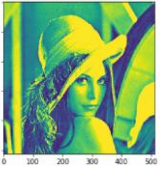
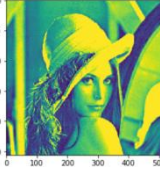

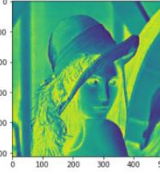
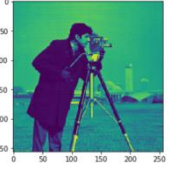

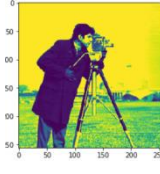
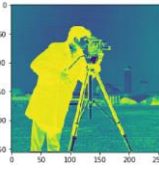
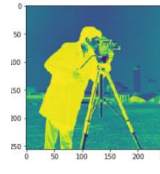
Image	Original	Brightness		Image Negation	
		PDT	CDT	PDT	CDT
(a)					
(b)					

Table 1: Comparison of Intensity Transformation under Paillier

5.1.4. Performance Analysis

MSE value zero means it is a perfect image. For brightness this algorithm performs perfectly but for negation there is a very small loss. The loss is not very big and is nearly perfect. But the time taken to encrypt and decrypt is very long.

Image	MSE	PSNR	SSIM	$t_{\text{encryption}}$	t_{apply}	$t_{\text{decryption}}$
(a)	0.0	infinity	1.0	341.841 s	1.267 s	299.372 s
(b)	0.0	infinity	1.0	27.967 s	0.113 s	23.612 s

Table : Image Brightness under Paillier

Image	MSE	PSNR	SSIM	$t_{\text{encryption}}$	t_{apply}	$t_{\text{decryption}}$
(a)	40.329	32.075	0.999	341.841 s	128.225 s	284.850 s
(b)	41.979	31.909	0.999	27.967 s	11.118 s	23.84 s

Table: Image Negation under Paillier

5.1.5. Key space Analysis

The secret key is 128-bits long, the key space is about 2^{128} . The encrypted image with a key size of 2^{128} is not easily affected by the brute force attack. Therefore this key size is sufficient. In hardware implementation the number of bits for the keys can be increased. However, by performing this the speed of the system may be decreased and volume of the hardware is increased

5.2 An Efficient Homomorphic Medical Image Encryption Algorithm

5.2.1. Histogram Analysis

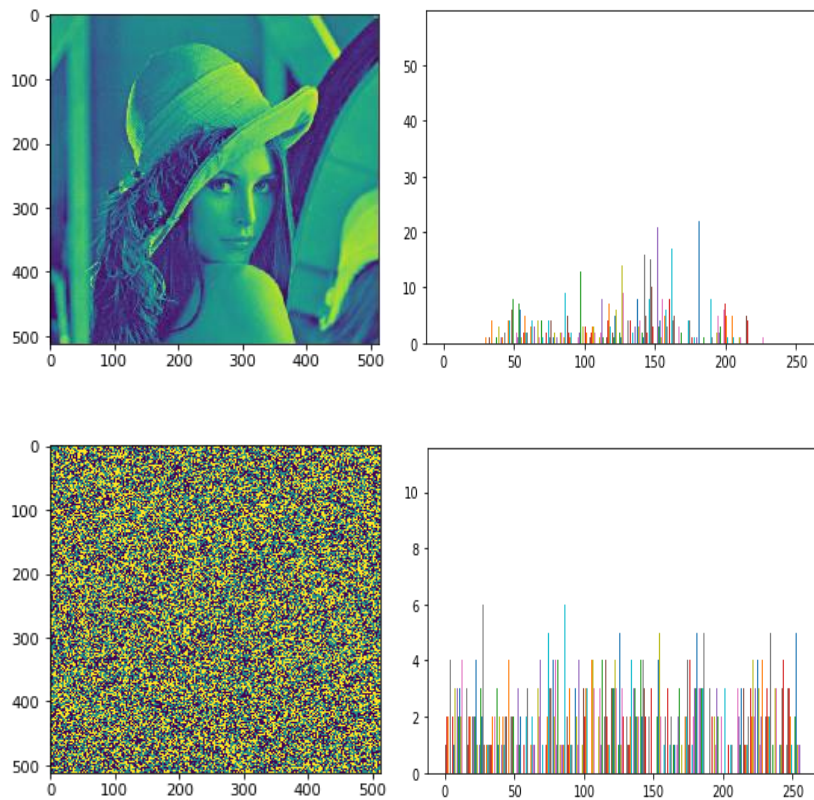


Fig: Histogram of Lena Image and Encrypted Lena Image for An Efficient Homomorphic Medical Image Encryption Algorithm

5.2.2 Correlation Analysis

5.2.2.1. Horizontal Correlation

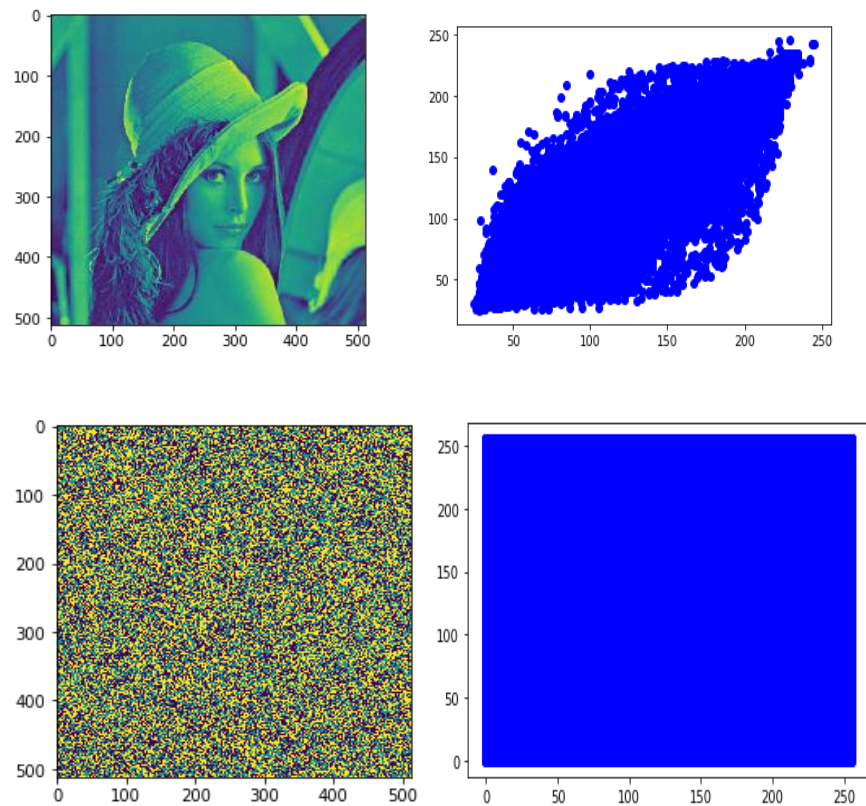
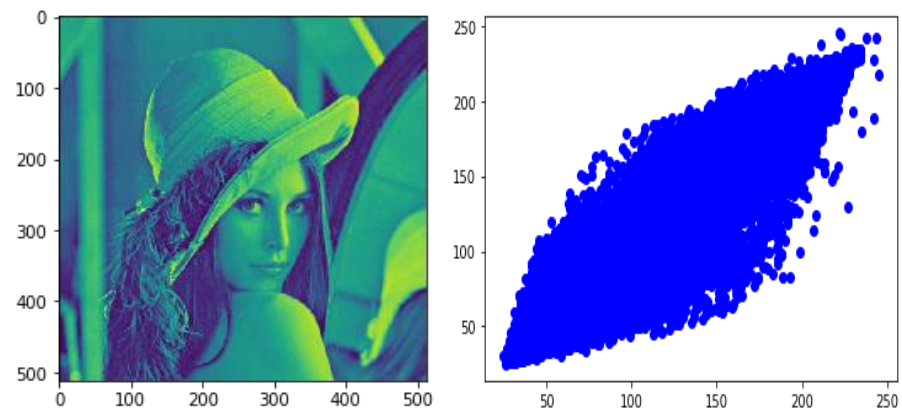


Fig: Horizontal correlation of Lena Image and Encrypted Lena image for An Efficient Homomorphic Medical Image Encryption Algorithm

5.2.2.2 Vertical Correlation



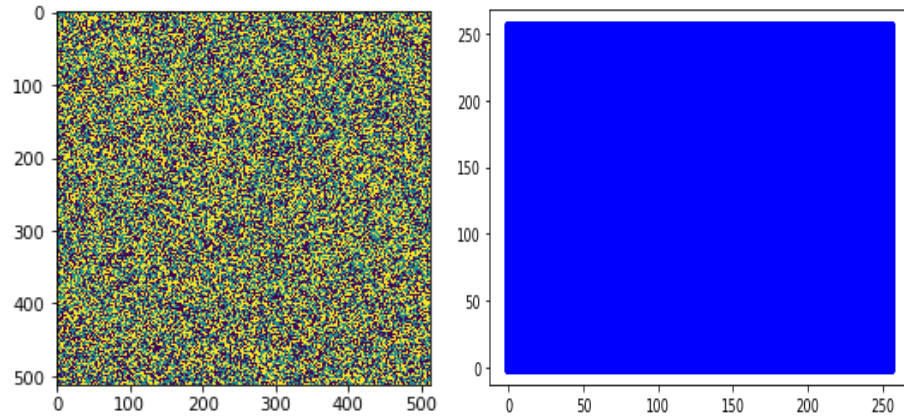


Fig: Vertical Correlation of Lena Image and Encrypted Lena image for An Efficient Homomorphic Medical Image Encryption Algorithm

5.2.3. Intensity Transformation

Image	Original	Brightness PDT CDT		Image Negation PDT CDT	
(a)					
(b)					

Table 1: Comparison of Intensity Transformation under An Efficient Homomorphic Medical Image Encryption Algorithm

5.2.4. Performance Analysis

MSE value zero means it is a perfect image. For brightness this algorithm performs perfectly but for negation there is some loss. Although from a visual point of view it is not noticeable. The time taken to encrypt and decrypt is very small compared to Paillier.

Image	MSE	PSNR	SSIM	t _{encryption}	t _{apply}	t _{decryption}
(a)	0.0	infinity	1.0	0.4717 s	0.1189 s	0.7629 s
(b)	0.0	infinity	1.0	0.1147 s	0.0359 s	0.1795 s

Table : Image Brightness under An Efficient Homomorphic Medical Image Encryption Algorithm

Image	MSE	PSNR	SSIM	t _{encryption}	t _{apply}	t _{decryption}
(a)	2479.4502	14.1872	0.9081	0.4717 s	0.1755 s	0.7549 s
(b)	256.3479	24.0425	0.7794	0.1147 s	0.0548 s	0.1778 s

Table : Image Negation under An Efficient Homomorphic Medical Image Encryption Algorithm

5.2.5. Key space Analysis

The secret key is 150-bits long, the key space is about 2^{150} . The encrypted image with a key size of 2^{150} is not easily affected by the brute force attack. Therefore this key size is sufficient. In hardware implementation the number of bits for the keys can be increased. However, by performing this the speed of the system may be decreased and volume of the hardware is increased.

6. Conclusion

In this project, we have performed a comparative analysis between two homomorphic image encryption algorithms: Paillier and Medical Image Encryption Algorithm(MIEA). We have done 5 analyses: Key space analysis, Histogram analysis, Correlation Analysis, Intensity Transformation Analysis and Performance analysis. Based on the key space analysis, both algorithms we implemented provide a sufficient key space for encryption and could also be scaled to a larger key space. From the histogram and correlation analysis we can infer that both the algorithms provide a good encryption scheme for randomness in cipher text. Both these algorithms are safe from image scaling attacks. From the Intensity Transformation and Performance Analysis we can infer that Paillier produces more accurate transformations but the time taken for encryption and decryption is large compared to MIEA. Even though MIEA is very fast, the error is too much when image processing is applied.

7. References

- [1] Ibtihal, Mouhib & El Ouadghiri, Driss & Hassan, Naanani. Homomorphic Encryption as a Service for Outsourced Images in Mobile Cloud Computing Environment. International Journal of Cloud Applications and Computing, (2017).
- [2] M. T. I. Ziad, A. Alanwar, M. Alzantot and M. Srivastava, "Cryptolmg: Privacy preserving processing over encrypted images," 2016 IEEE Conference on Communications and Network Security (CNS), Philadelphia, PA, 2016
- [3] Sachin Rana, Om Jadhav , Shivam Rajput, Pranjal Bhansali, Varshapriya Jyotinagar, Homomorphic Image Encryption, International Research Journal of Engineering and Technology (IRJET) Vol 6 Issue 4, Apr 2019
- [4] Pan Yang, Xiaolin Gui, Jian An, and Feng Tian, An Efficient Secret Key Homomorphic Encryption Used in Image Processing Service, Security and Communication Networks Volume (2017)
- [5] Wade, M. I., Chouikha, M., Gill, T., Patterson, W., Washington, T. M., & Zeng, J. (2019, October). Distributed Image Encryption Based On a Homomorphic Cryptographic Approach. In 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON) (pp. 0686-0696). IEEE.
- [6] Xiong, L., & Dong, D. (2019). Reversible data hiding in encrypted images with somewhat homomorphic encryption based on sorting block-level prediction-error expansion. *Journal of Information Security and Applications*, 47, 78-85.
- [7] Li Li , Ahmed A. Abd El-Latif , Xiamu Niu ,Elliptic curve ElGamal based homomorphic image encryption scheme for sharing secret images Signal Processing 92 (2012) ,1069–1078 Elsevier
- [8] Mamadoul Wade , HenryC. Ogworonjo , Madiha Gul Mandoye Ndoeye , Mohamed Chouikha , Wayne Patterson Red Green Blue Image Encryption Based on Paillier Cryptographic System San Jose Conference 2018
- [9] Mohsin Shah, Weiming Zhang, Honggang Hu, and Nenghai Yu. 2019. Paillier Cryptosystem based Mean Value Computation for Encrypted Domain Image Processing Operations. ACM Trans. Multimedia Comput. Commun. Appl. 15, 3, Article 76 (September 2019)
- [10] Ritu Gupta, Anurag Mishra, Sarika Jain . Secure Image Watermarking in a Compressed SPIHT Domain Using Paillier Cryptosystem. International Journal of Information System Modeling and Design Volume 10 • Issue 4 • October-December 2019